Symfony Messenger Workshop

SymfonyCon Lisbon

Samuel Rozé

VP Engineering @ Birdie

Symfony
ContinuousPipe
ApiPlatform

Workshop's master plan

Workshop's master plan

1. Brief introduction to Symfony Messenger

Workshop's master plan

1. Brief introduction to Symfony Messenger
2. We code.

What's Symfony Messenger about?

What's Symfony Messenger about?

1. **Messages**. Any PHP object.

What's Symfony Messenger about?

1. **Messages**. Any PHP object.
2. **Message bus**. Where you dispatch your messages.

What's Symfony Messenger about?

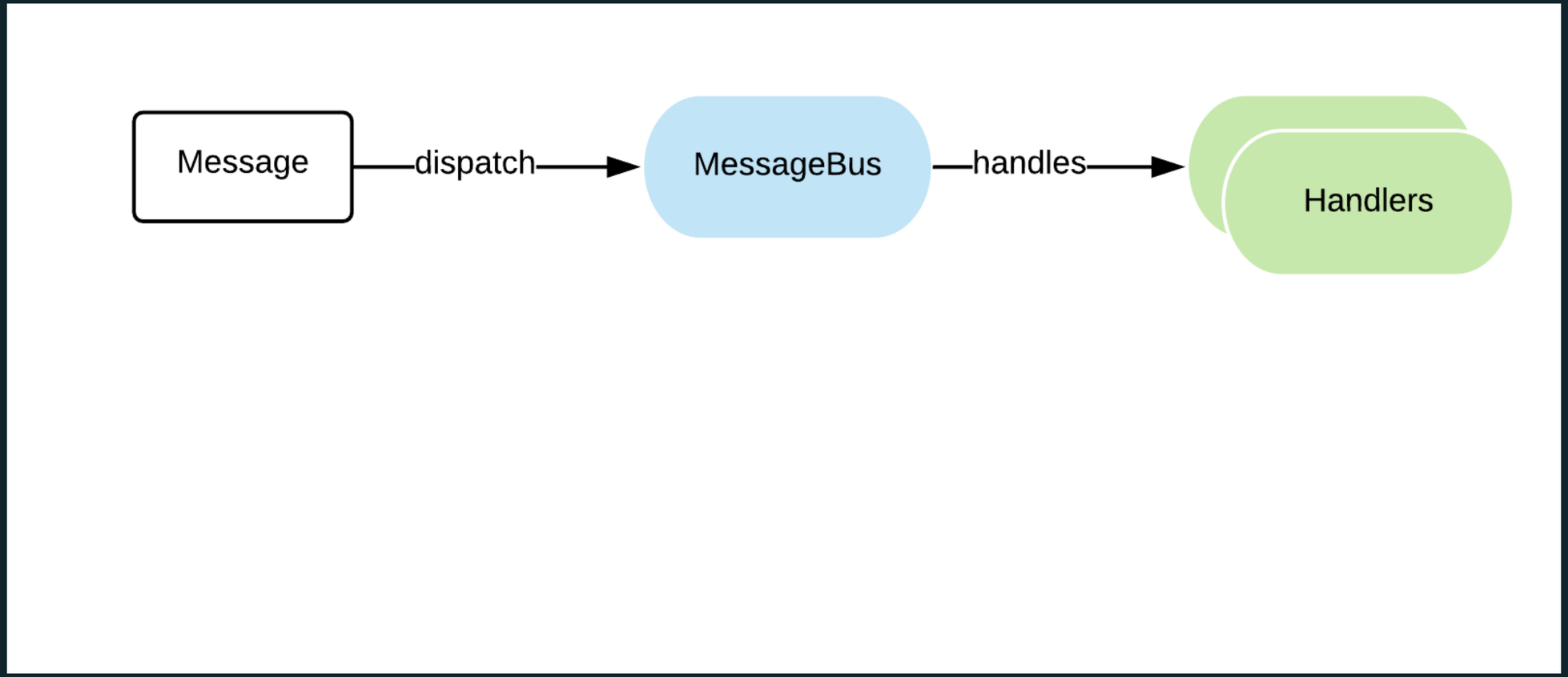
1. **Messages**. Any PHP object.
2. **Message bus**. Where you dispatch your messages.
3. **Message handlers**. Will execute your business logic when the message arrives to them.

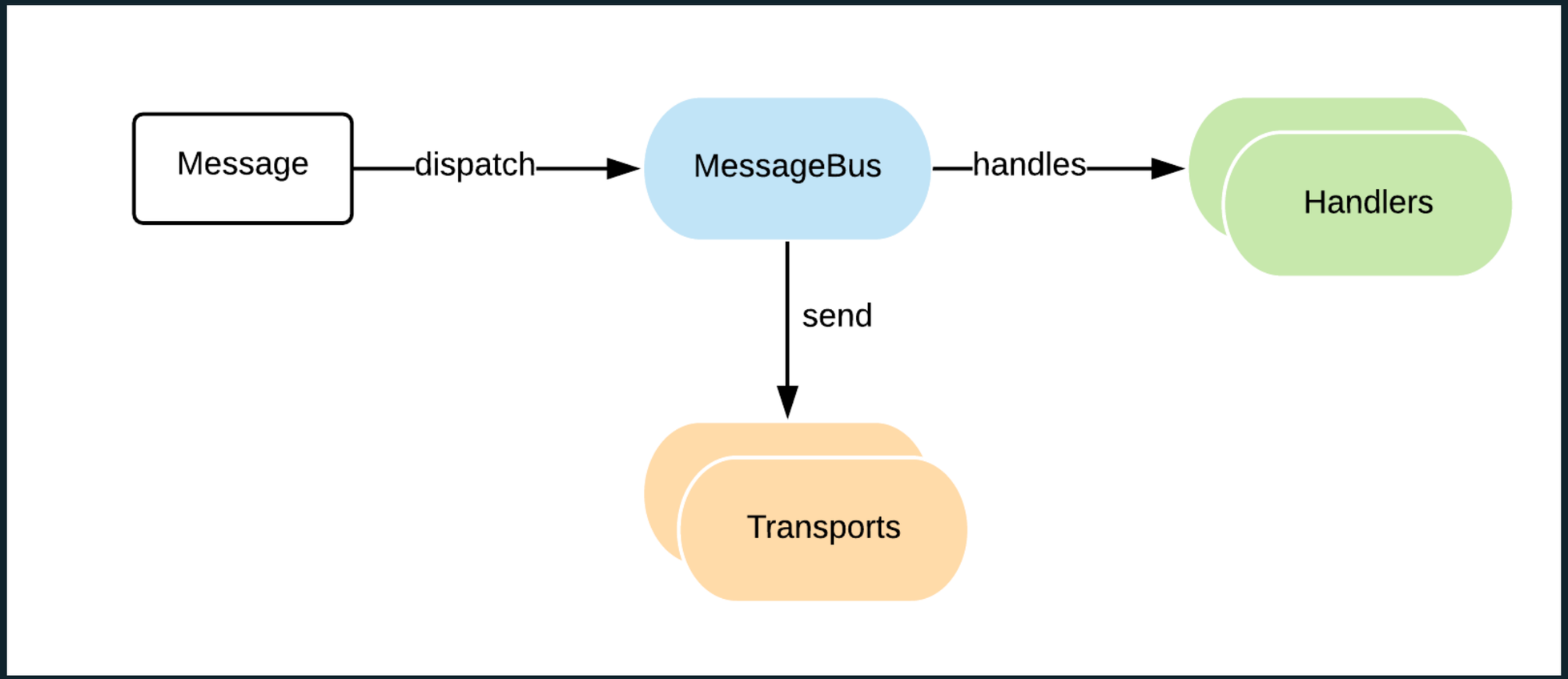
What's Symfony Messenger about?

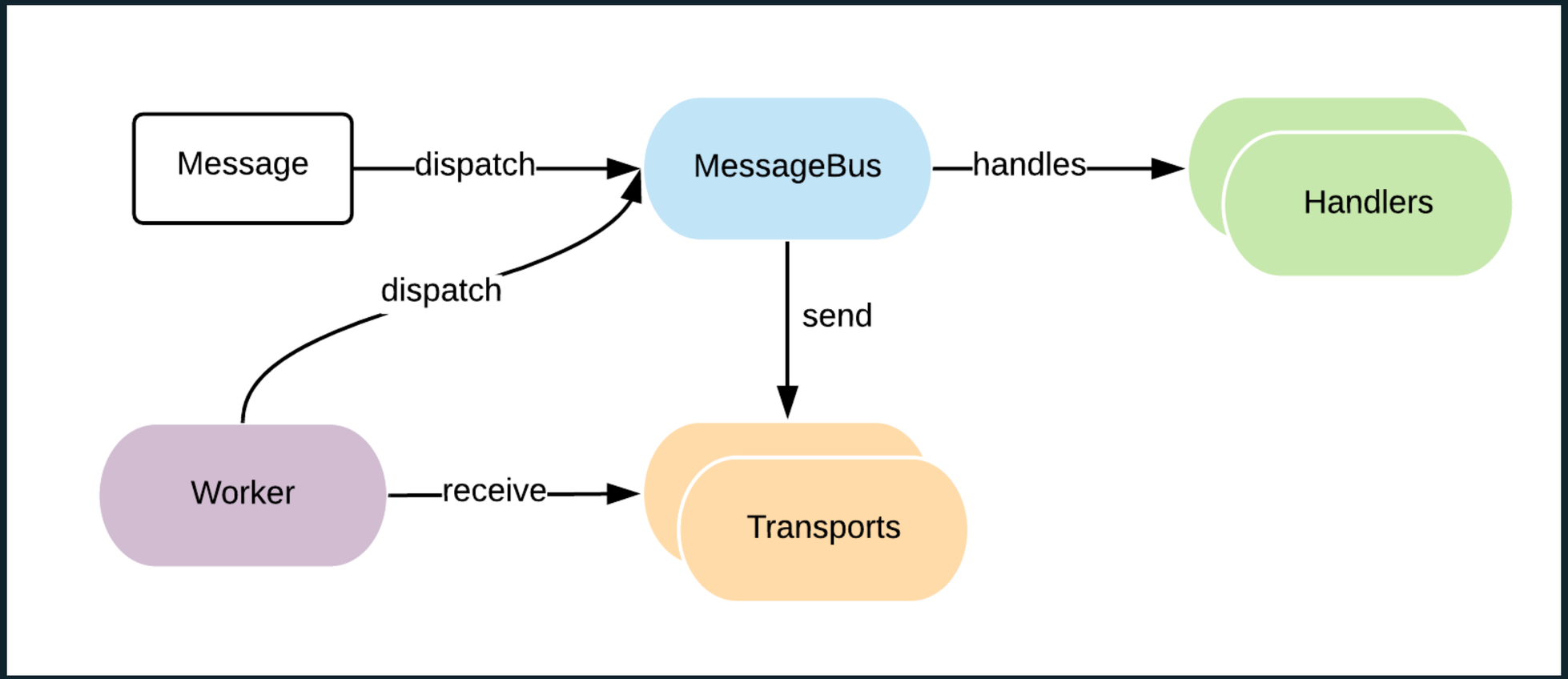
1. **Messages**. Any PHP object.
2. **Message bus**. Where you dispatch your messages.
3. **Message handlers**. Will execute your business logic when the message arrives to them.
4. **Transports**. Allow to send and receive messages through 3rd party systems.

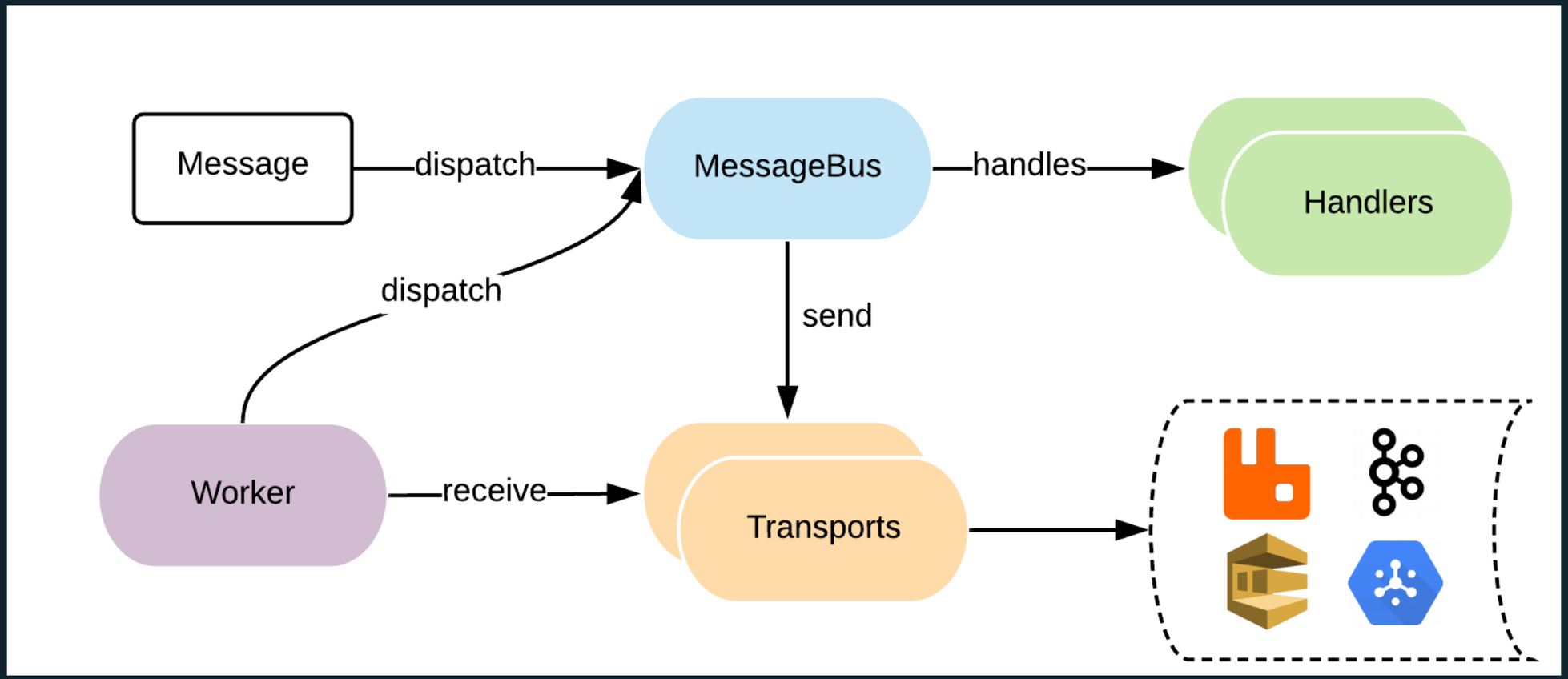
What's Symfony Messenger about?

1. **Messages**. Any PHP object.
2. **Message bus**. Where you dispatch your messages.
3. **Message handlers**. Will execute your business logic when the message arrives to them.
4. **Transports**. Allow to send and receive messages through 3rd party systems.
5. **Worker.** To consume messages from transports.


Message
dispatch
MessageBus
handles
Handlers


Message
dispatch
MessageBus
handles
Handlers
send
Transports


Message
dispatch
MessageBus
handles
Handlers
dispatch
send
Worker
receive
Transports


Message
dispatch
MessageBus
handles
Handlers
dispatch
send
Worker
receive
Transports

A message

No specific requirement.

```php
namespace App\Message;

class SendNotification
{
    // ...properties

    public function __construct(string $message, array $users)
    {
        $this->message = $message;
        $this->users = $users;
    }

    // ...getters
}
```

As a component

```php
use Symfony\Component\Messenger\MessageBus;
use Symfony\Component\Messenger\HandlerLocator;
use Symfony\Component\Messenger\Middleware\HandleMessageMiddleware;

$handler = function(MyMessage $message) {
    // ...
};

$messageBus = new MessageBus([
    new HandleMessageMiddleware(new HandlerLocator([
        SendNotification::class => [ $handler ],
    ])),
]);
```

In your Symfony application

Nothing to do...

Dispatching a message

```php
namespace App\Controller;

use Symfony\Component\Messenger\MessageBusInterface;
// ...

class DefaultController
{
    public function index(MessageBusInterface $bus, Request $request)
    {
        $bus->dispatch(new SendNotification(/* ... */));

        return new Response('<html><body>OK.</body></html>');
    }
}
```

Dispatching a message

```php
namespace App\Controller;

use Symfony\Component\Messenger\MessageBusInterface;
// ...

class DefaultController
{
    public function index(MessageBusInterface $bus, Request $request)
    {
        $bus->dispatch(new SendNotification(/* ... */));

        return new Response('<html><body>OK.</body></html>');
    }
}
```

No handler for message "App\Message\SendNotification".

Exception | Logs 1 | Stack Trace

Symfony\Component\Messenger\Exception\

NoHandlerForMessageException

in /Users/samuelroze/git/symfony/src/Symfony/Component/Messenger/**ContainerHandlerLocator.php** (line 36)

```
31.     {
32.         $messageClass = get_class($message);
33.         $handlerKey = 'handler.'.$messageClass;
34.
35.         if (!$this->container->has($handlerKey)) {
36.             throw new NoHandlerForMessageException(sprintf('No handler for message "%s".', $messageClass));
37.         }
38.
39.         return $this->container->get($handlerKey);
40.     }
41. }
```

ContainerHandlerLocator->**resolve**(*object*(SendNotification))
in /Users/samuelroze/git/symfony/src/Symfony/Component/Messenger/Middleware/**HandleMessageMiddleware.php** (line 34)

HandleMessageMiddleware->**handle**(*object*(SendNotification), *object*(Closure))
in /Users/samuelroze/git/symfony/src/Symfony/Component/Messenger/**MessageBus.php** (line 56)

A message handler

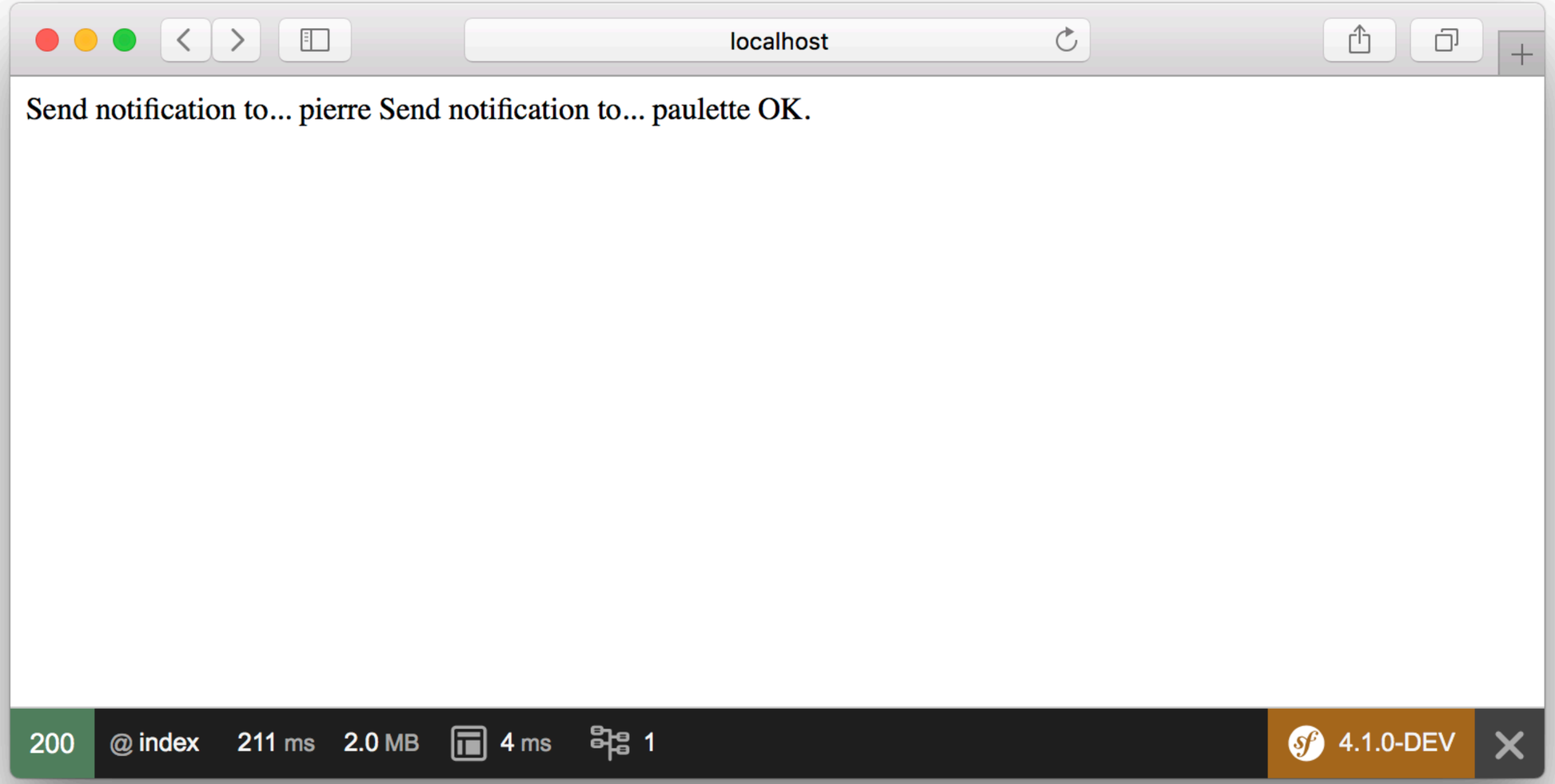
```php
namespace App\MessageHandler;

use App\Message\SendNotification;

class SendNotificationHandler
{
    public function __invoke(SendNotification $message)
    {
        foreach ($message->getUsers() as $user) {
            echo "Send notification to... ".$user."\n";
        }
    }
}
```

Register your handler

```yaml
# config/services.yaml
services:
    App\MessageHandler\SendNotificationHandler:
        tags:
            - messenger.message_handler
```

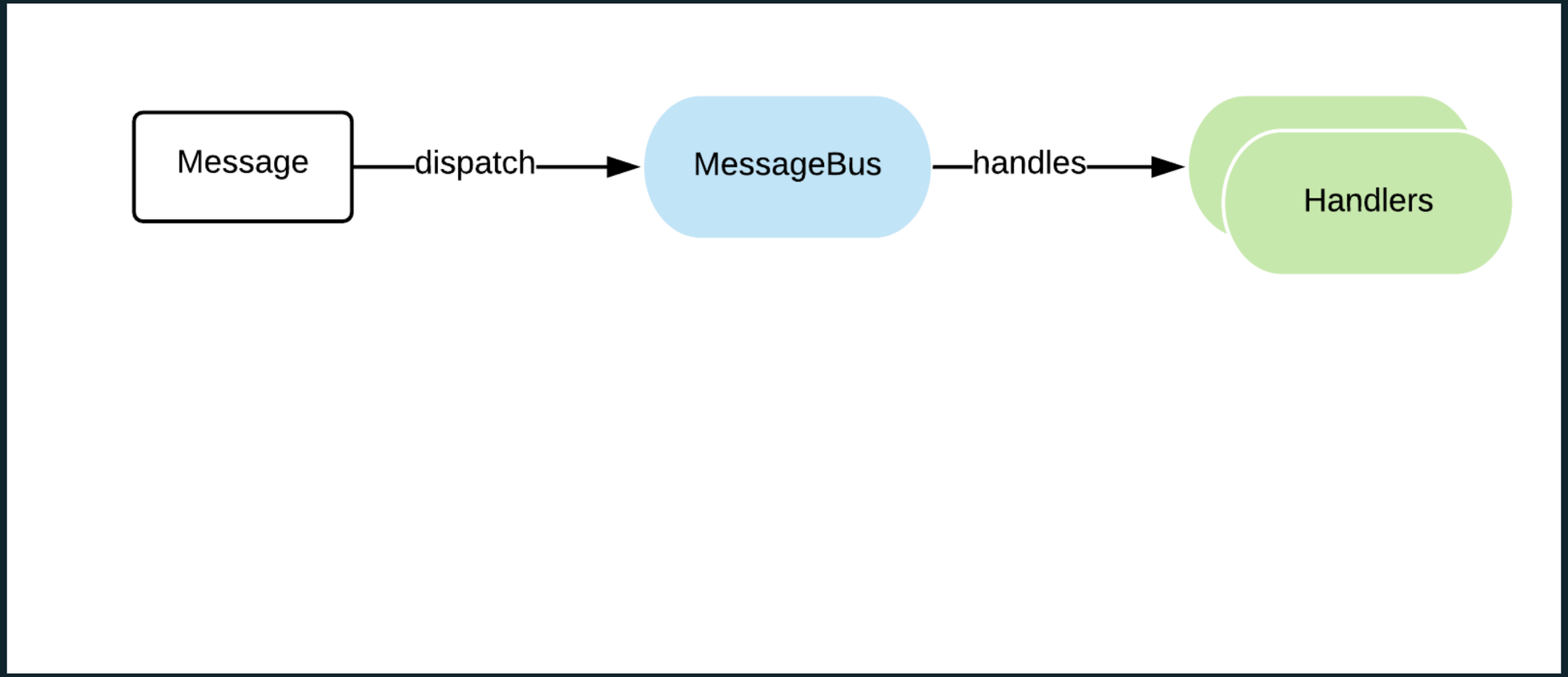
Register your handlers (option 2)

```yaml
# config/services.yaml
services:
    App\MessageHandler\:
        resource: '../src/MessageHandler/*'
        tags: ['messenger.message_handler']
```

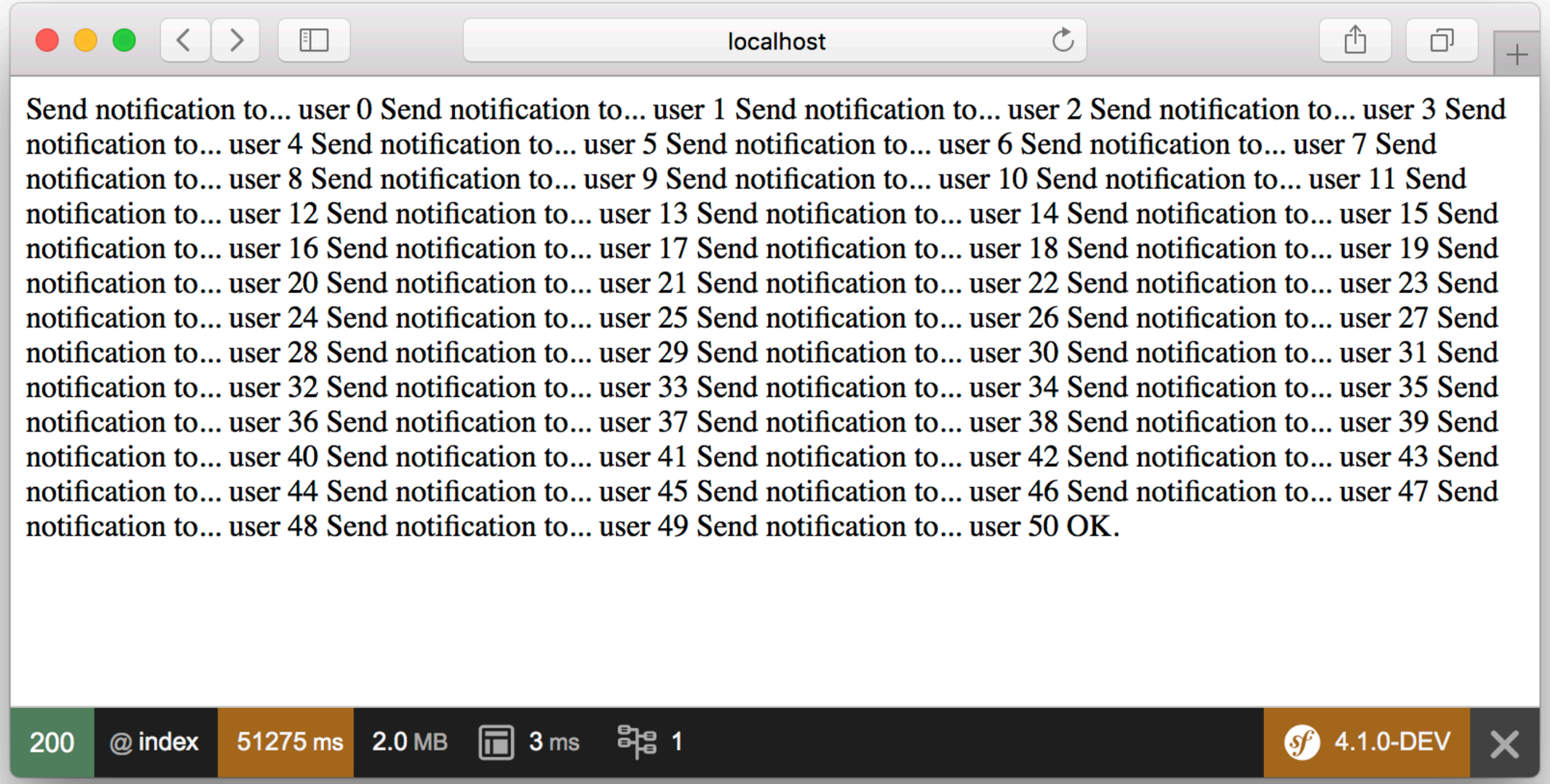

localhost
Send notification to... pierre Send notification to... paulette OK.
200
@ index
211 ms
2.0 MB
4 ms
1
4.1.0-DEV

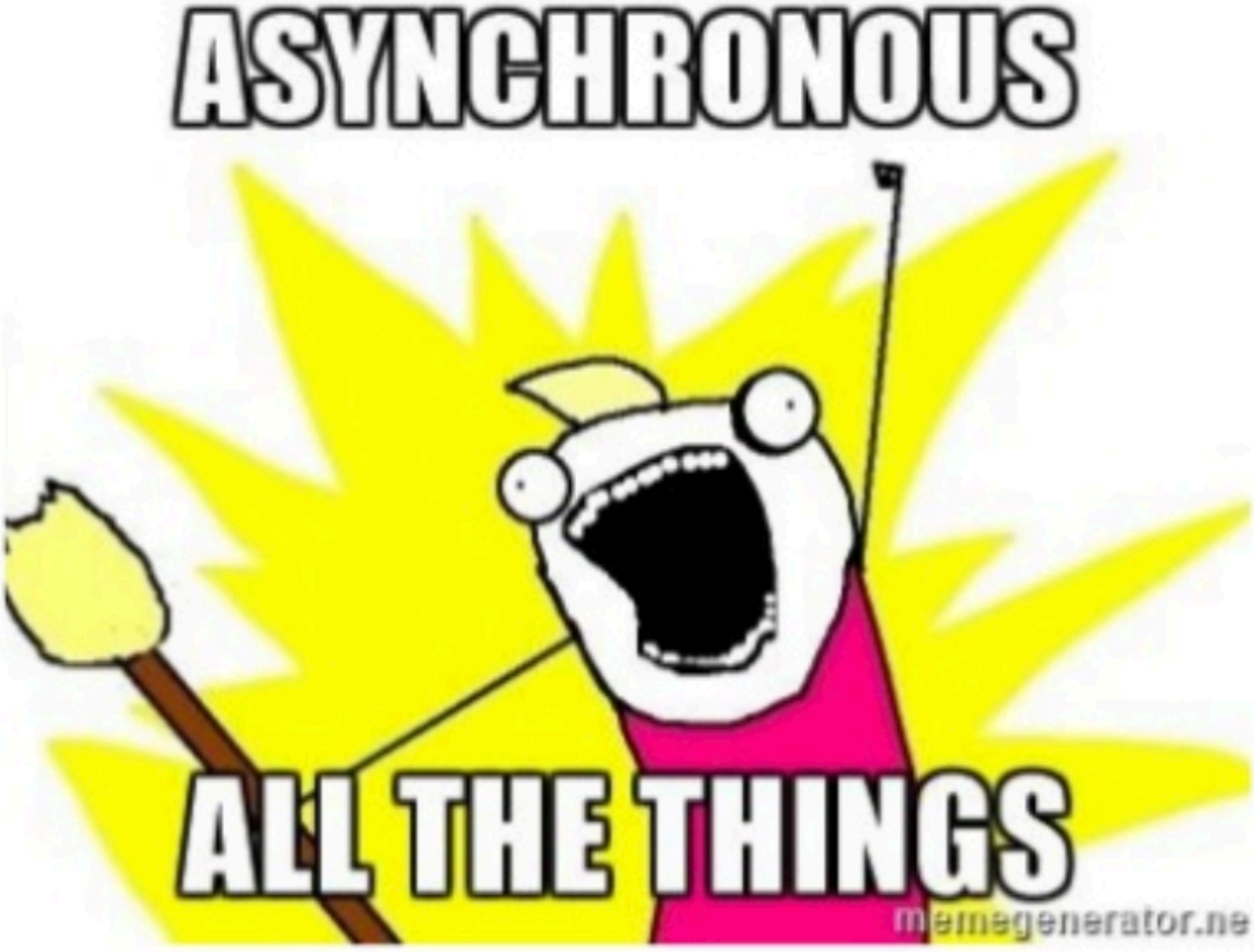
Yay, dispatched !

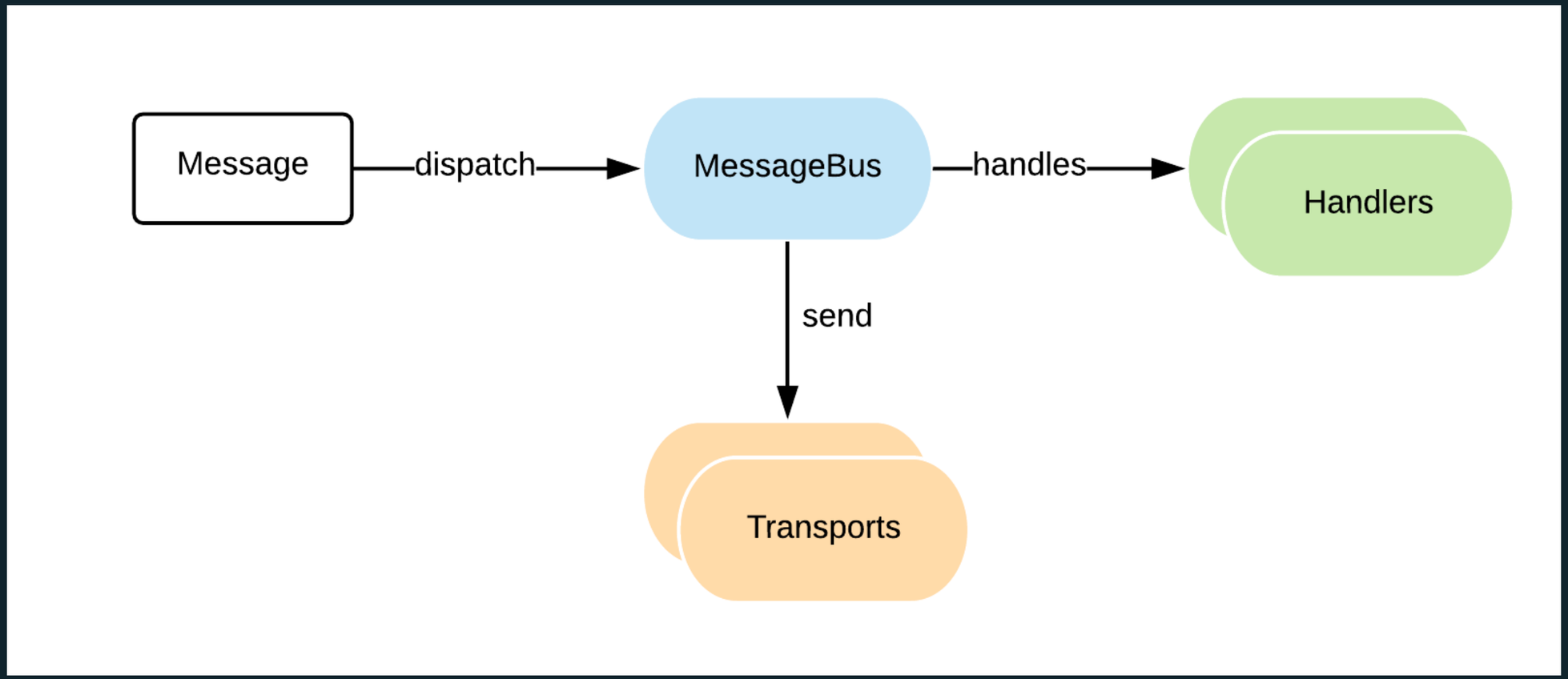
Handler is called. Done.


Message
dispatch
MessageBus
handles
Handlers

Now, imagine your handler takes ages to run.


localhost
Send notification to... user 0 Send notification to... user 1 Send notification to... user 2 Send notification to... user 3 Send notification to... user 4 Send notification to... user 5 Send notification to... user 6 Send notification to... user 7 Send notification to... user 8 Send notification to... user 9 Send notification to... user 10 Send notification to... user 11 Send notification to... user 12 Send notification to... user 13 Send notification to... user 14 Send notification to... user 15 Send notification to... user 16 Send notification to... user 17 Send notification to... user 18 Send notification to... user 19 Send notification to... user 20 Send notification to... user 21 Send notification to... user 22 Send notification to... user 23 Send notification to... user 24 Send notification to... user 25 Send notification to... user 26 Send notification to... user 27 Send notification to... user 28 Send notification to... user 29 Send notification to... user 30 Send notification to... user 31 Send notification to... user 32 Send notification to... user 33 Send notification to... user 34 Send notification to... user 35 Send notification to... user 36 Send notification to... user 37 Send notification to... user 38 Send notification to... user 39 Send notification to... user 40 Send notification to... user 41 Send notification to... user 42 Send notification to... user 43 Send notification to... user 44 Send notification to... user 45 Send notification to... user 46 Send notification to... user 47 Send notification to... user 48 Send notification to... user 49 Send notification to... user 50 OK.
200
@ index
51275 ms
2.0 MB
3 ms
1
4.1.0-DEV


ASYNCHRONOUS
ALL THE THINGS
memegenerator.ne


Message
dispatch
MessageBus
handles
Handlers
send
Transports

What's a transport?

What's a transport?

1. **Send & Receive**.
   Sends and receives… messages.

What's a transport?

1. **Send & Receive**.
   Sends and receives… messages.
2. **Is configurable via a DSN**
   So we have a unique and common way of configuring them.

What's a transport?

1. **Send & Receive**.
   Sends and receives… messages.
2. **Is configurable via a DSN**
   So we have a unique and common way of configuring them.
3. **Uses a serializer**
   By defaut, uses Symfony Serializer. Replace as you wish.

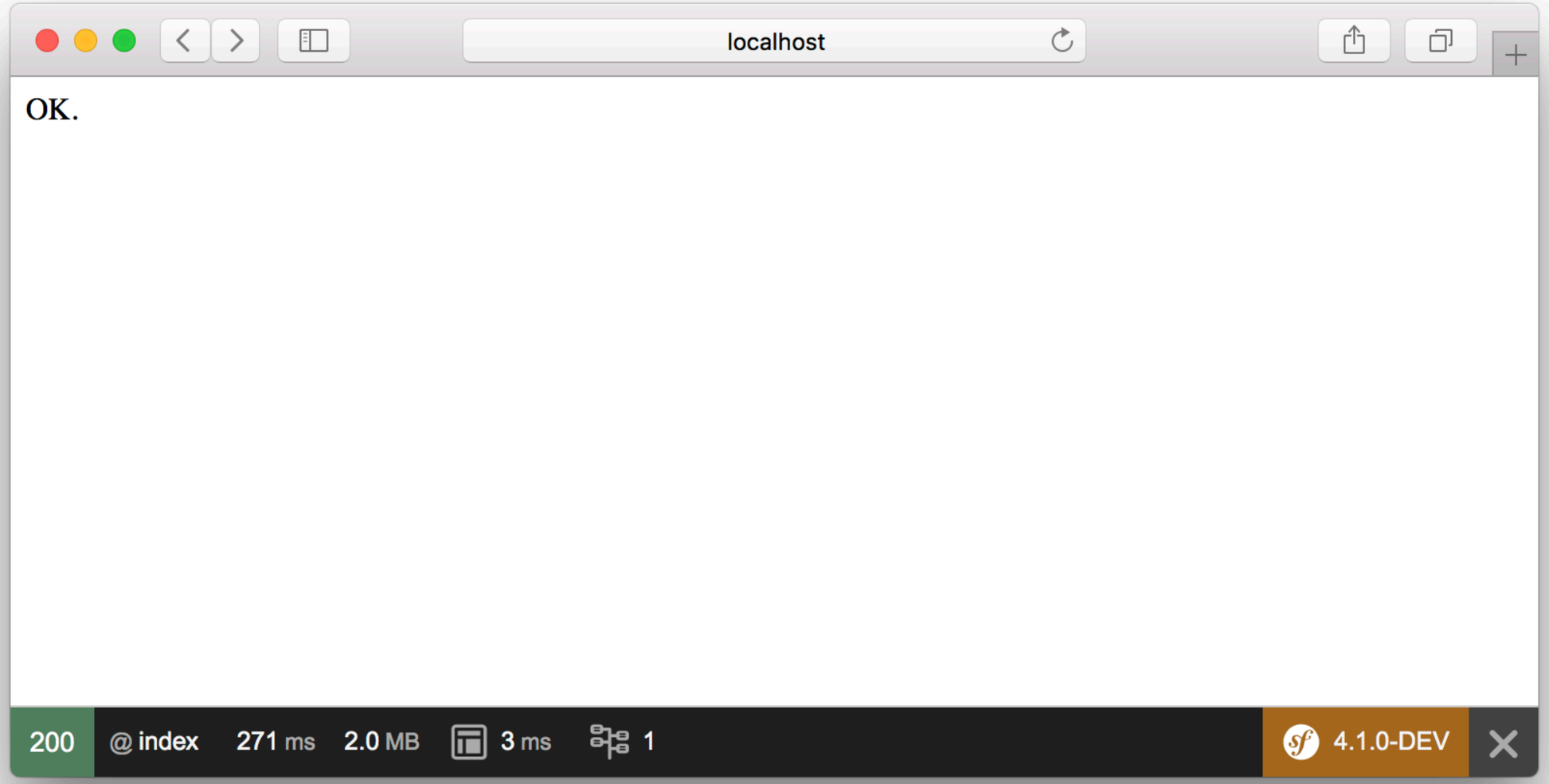
Configure your transport(s)

```yaml
# config/packages/messenger.yaml
framework:
    messenger:
        transports:
            default: '%env(MESSENGER_DSN)%'
```

```
# .env
MESSENGER_DSN=amqp://guest:guest@localhost:5672/%2f/messages
```

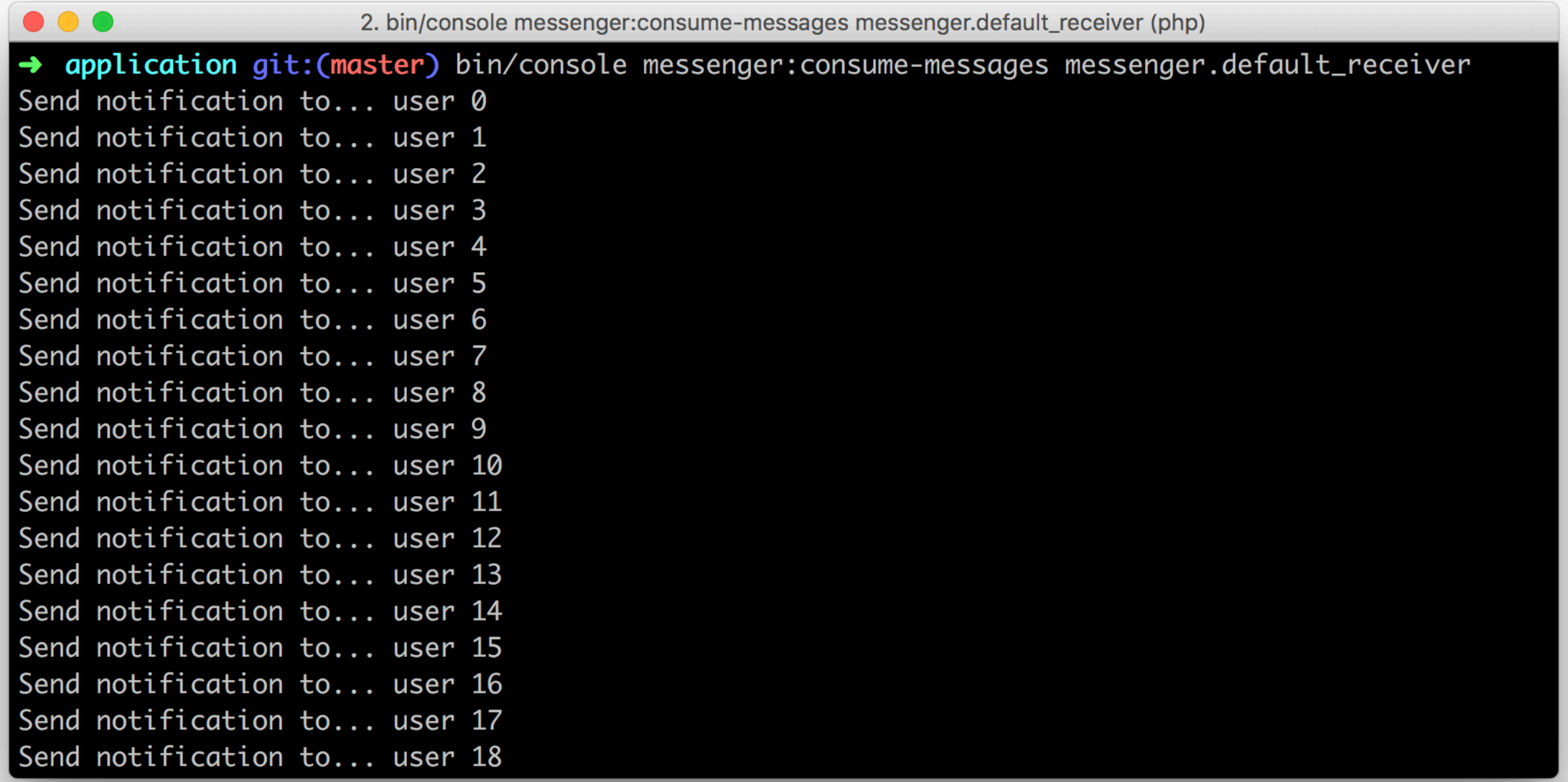
Route your message(s)

```
framework:
    messenger:
        routing:
            'App\Message\SendNotification': default
```


localhost
OK.
200
@ index
271 ms
2.0 MB
3 ms
1
4.1.0-DEV


BUT... IT DIDN'T
DO ANYTHING, RIGHT?

```
$ bin/console messenger:consume-messages
```


2. bin/console messenger:consume-messages messenger.default_receiver (php)
➜ application git:(master) bin/console messenger:consume-messages messenger.default_receiver
Send notification to... user 0
Send notification to... user 1
Send notification to... user 2
Send notification to... user 3
Send notification to... user 4
Send notification to... user 5
Send notification to... user 6
Send notification to... user 7
Send notification to... user 8
Send notification to... user 9
Send notification to... user 10
Send notification to... user 11
Send notification to... user 12
Send notification to... user 13
Send notification to... user 14
Send notification to... user 15
Send notification to... user 16
Send notification to... user 17
Send notification to... user 18

Multiple buses

```yaml
framework:
    messenger:
        buses:
            messenger.bus.commands: ~
            messenger.bus.events:
                middleware:
                    - App\\My\\Own\\Middleware
```

Envelopes

Add extra "non-domain" information to your messages.

Envelopes

```php
use Symfony\Component\Messenger\Envelope;
use Symfony\Component\Messenger\Stamp\SerializerStamp;

$bus->dispatch(
    (new Envelope($message))->with(new SerializerStamp([
        'groups' => ['my_serialization_groups'],
    ]))
);
```

Envelopes

```php
class MyOwnMiddleware implements MiddlewareInterface
{
    public function handle(Envelope $envelope, StackInterface $stack)
    {
        $envelopeItem = $envelope->last(AnotherStamp::class);

        // ...

        return $stack->next()->handle(
            $envelope->with(new AnotherStamp(/* ... */)),
            $stack
        );
    }
}
```

Envelopes

```php
class MyOwnMiddleware implements MiddlewareInterface
{
    public function handle($envelope, StackInterface $stack)
    {
        $envelopeItem = $envelope->last(AnotherStamp::class);

        // ...

        return $stack->next()->handle(
            $message->with(new AnotherStamp(/* ... */)),
            $stack
        );
    }
}
```

Transports

Transports

1. **Built-in AMQP**. You don't need anything more than the component and the PHP extension.

Transports

1. **Built-in AMQP**. You don't need anything more than the component and the PHP extension.
2. **Look at Enqueue**. Supports many cloud-based transports and Kafka, Gearman, etc... https://github.com/php-enqueue/messenger-adapter

Transports

1. **Built-in AMQP**. You don't need anything more than the component and the PHP extension.
2. **Look at Enqueue**. Supports many cloud-based transports and Kafka, Gearman, etc...
   https://github.com/php-enqueue/messenger-adapter
3. **Create your own.**
   Fully extendable by registering "factories".

Let's code!

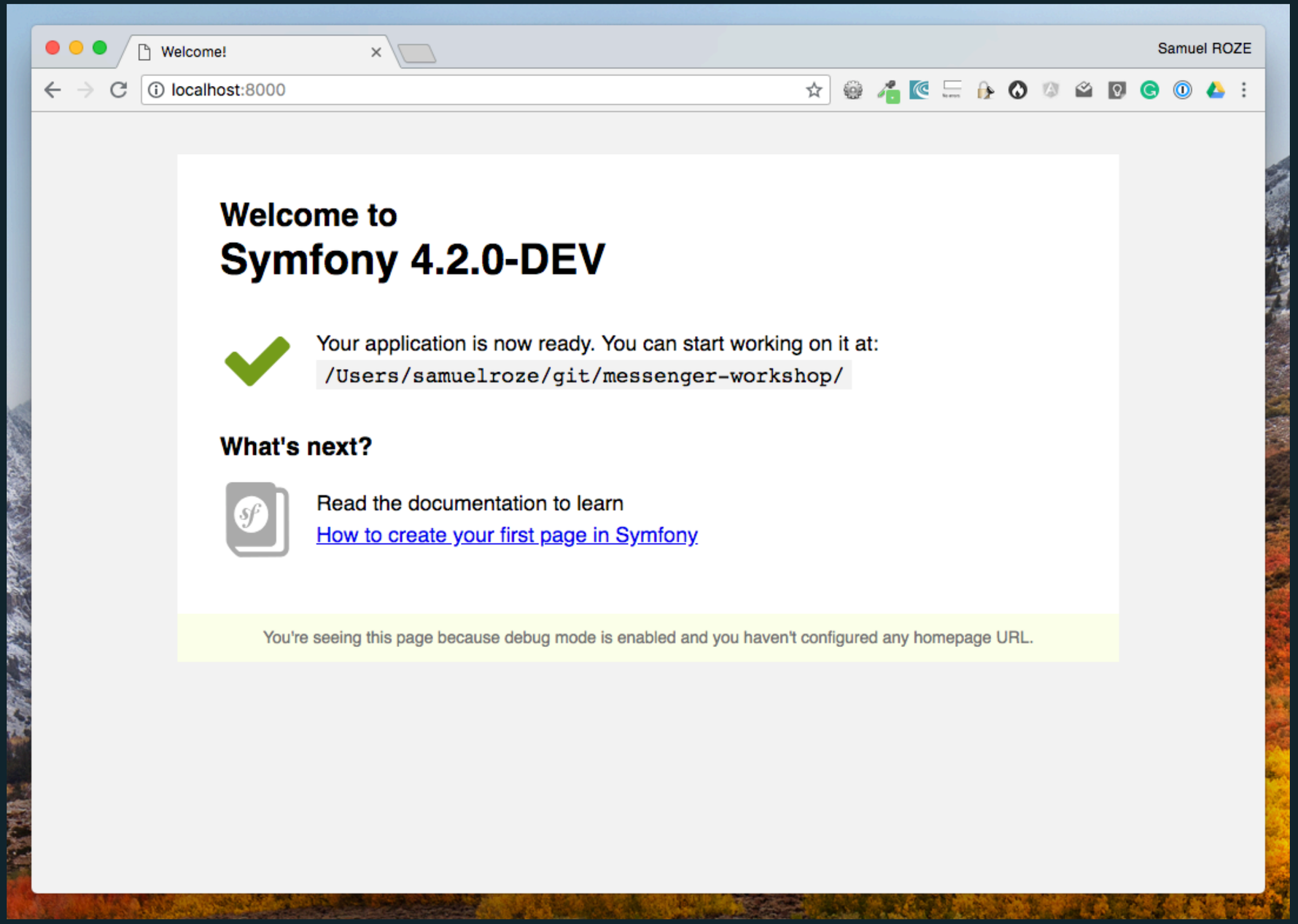
Our domain

A very simple application that will be able to track bets on football games. Via the user interface, we will:

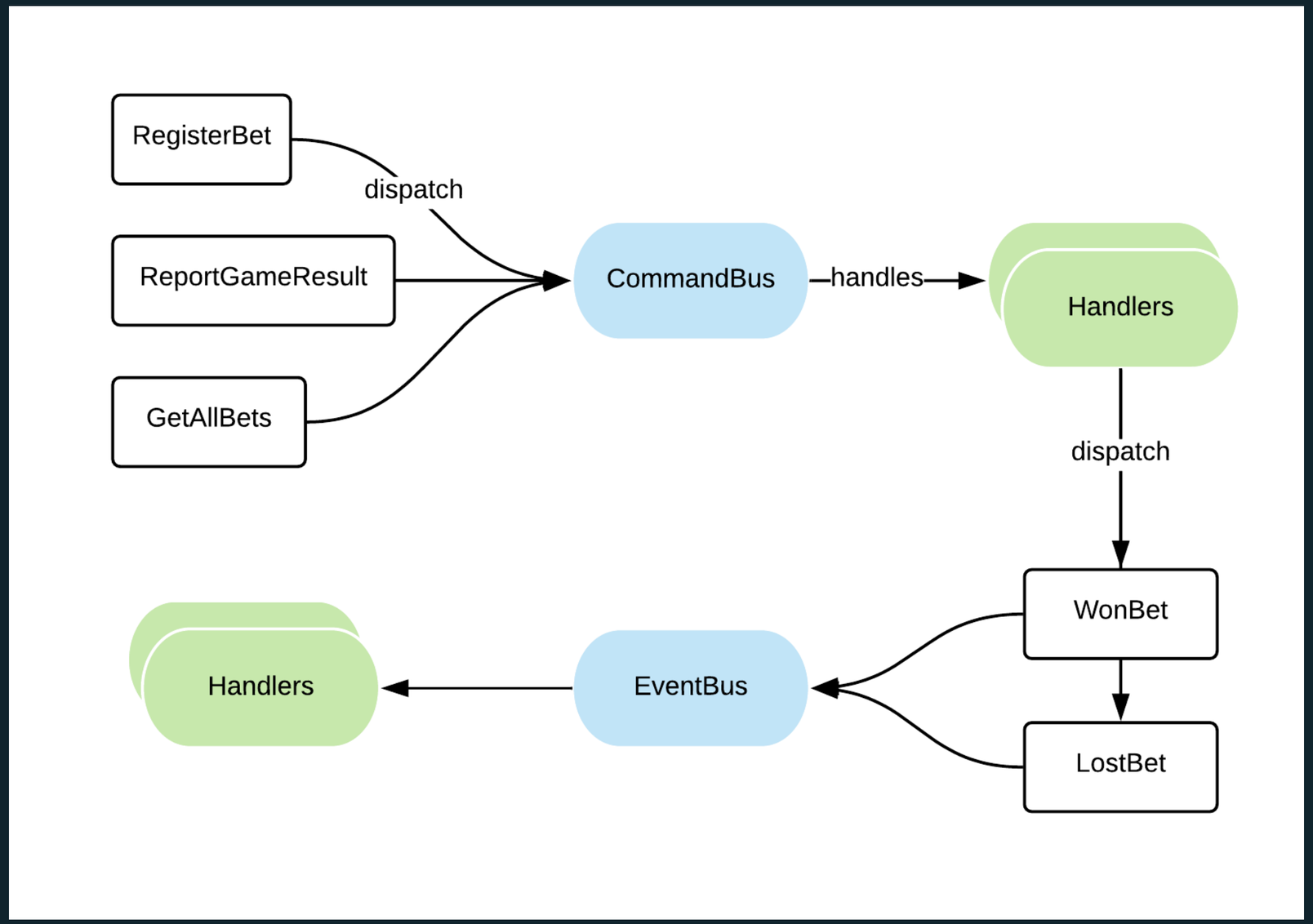
1. Register bets on different games
2. Submit the actual result of a game
3. Send a notification to the winners and losers

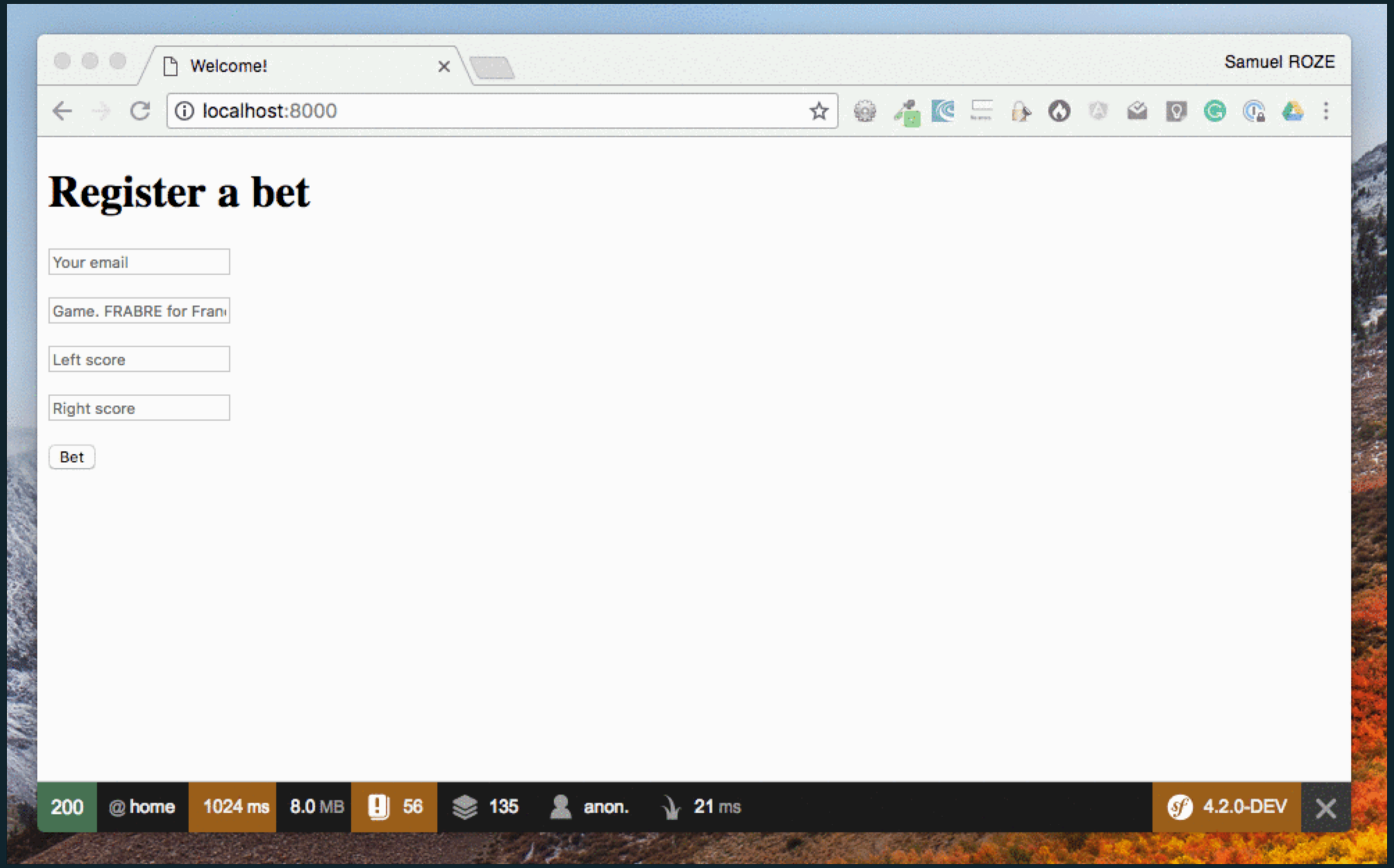
```
$ composer create-project symfony/website-skeleton
messenger-workshop
```

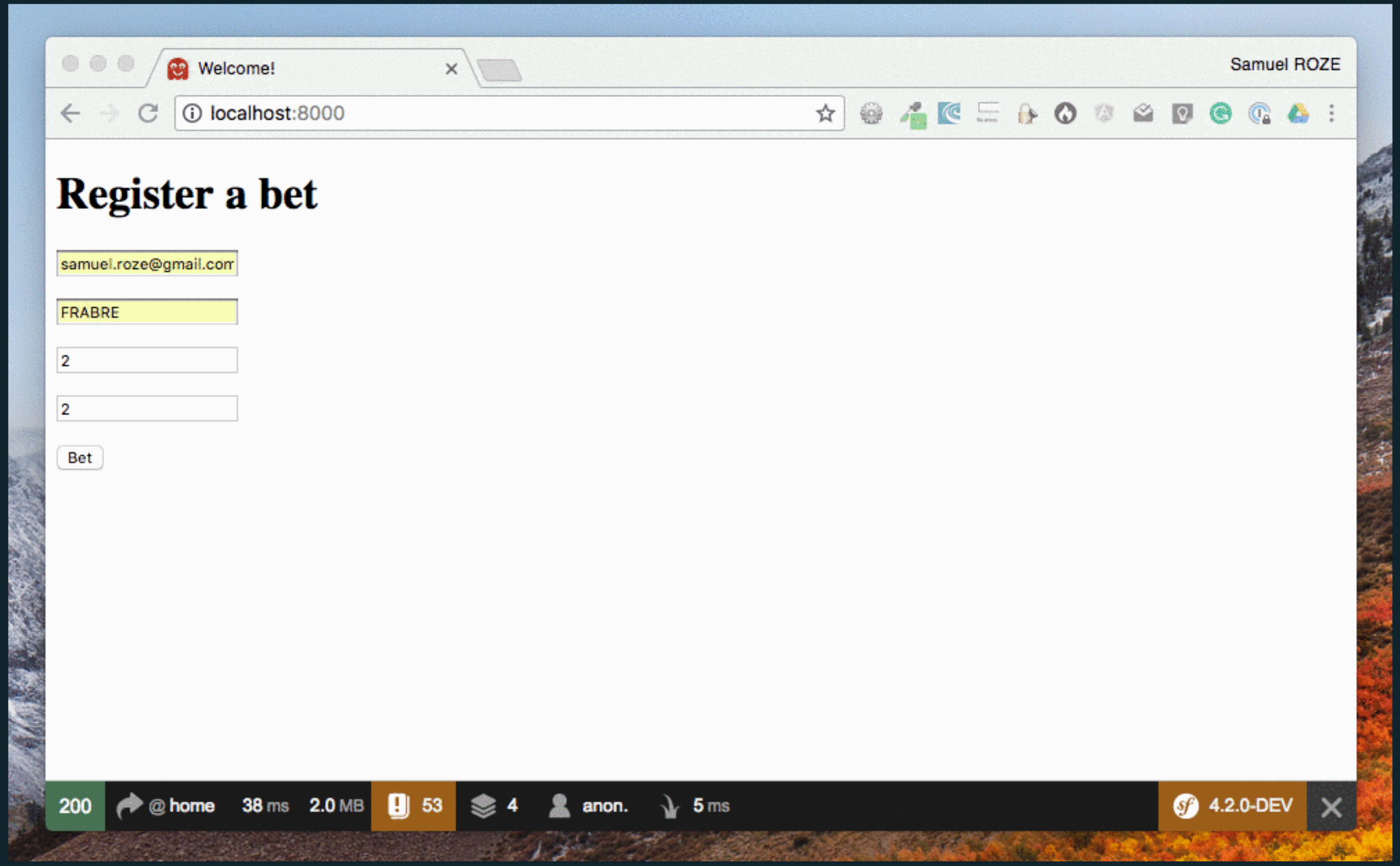
```
$ bin/console server:run
```

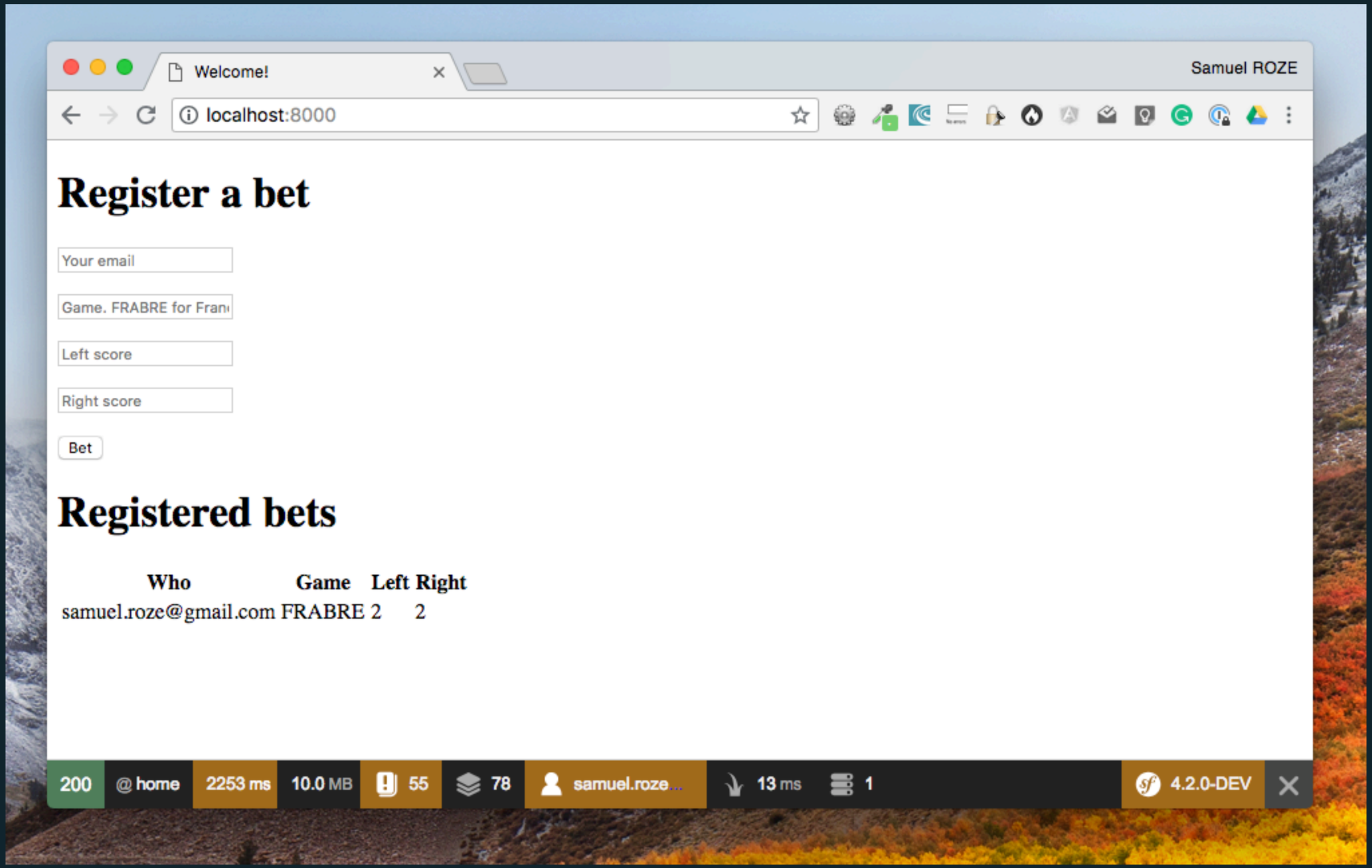

Welcome!
Samuel ROZE
localhost:8000
Welcome to
Symfony 4.2.0-DEV
Your application is now ready. You can start working on it at:
/Users/samuelroze/git/messenger-workshop/
What's next?
Read the documentation to learn
How to create your first page in Symfony
You're seeing this page because debug mode is enabled and you haven't configured any homepage URL.

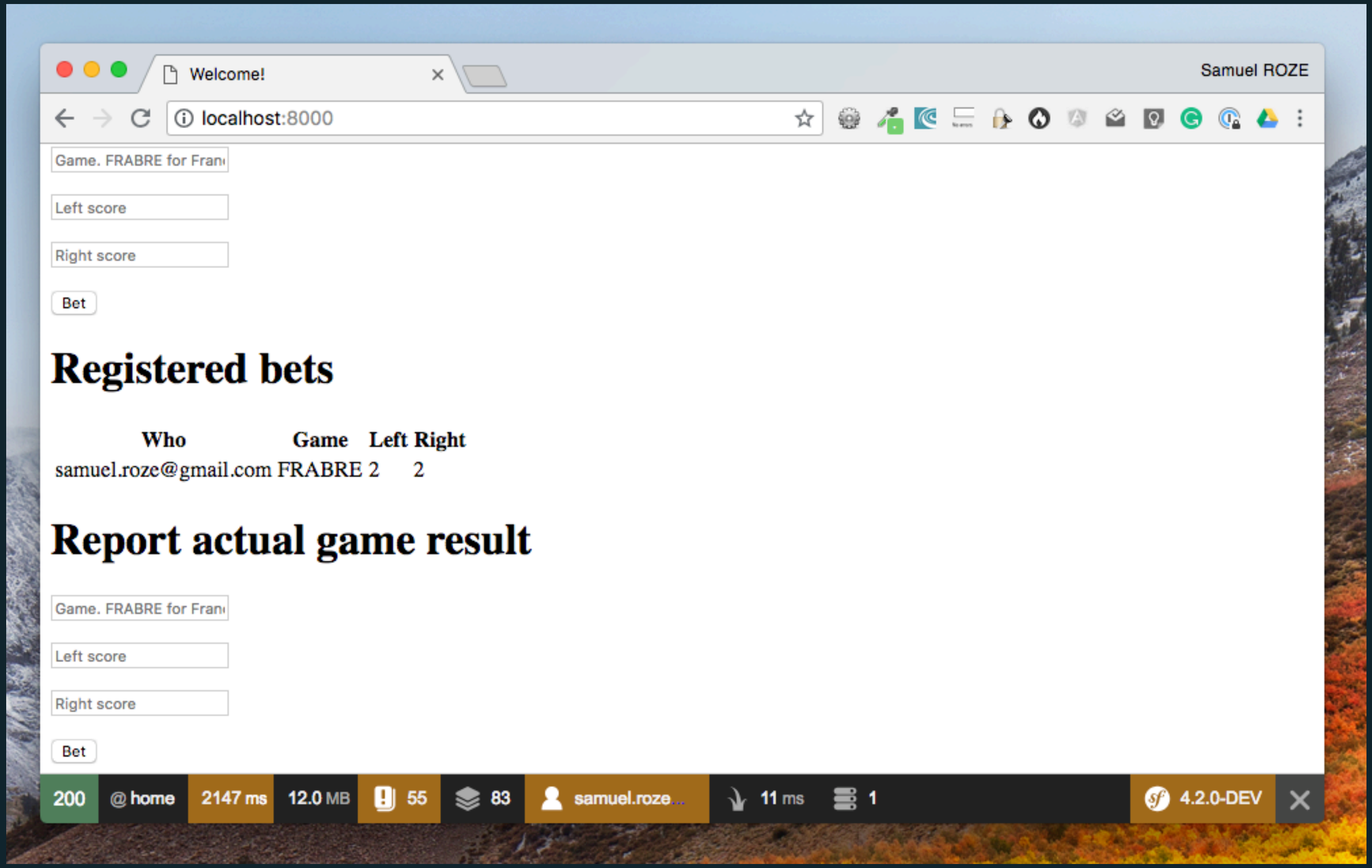
```
$ composer req messenger
```

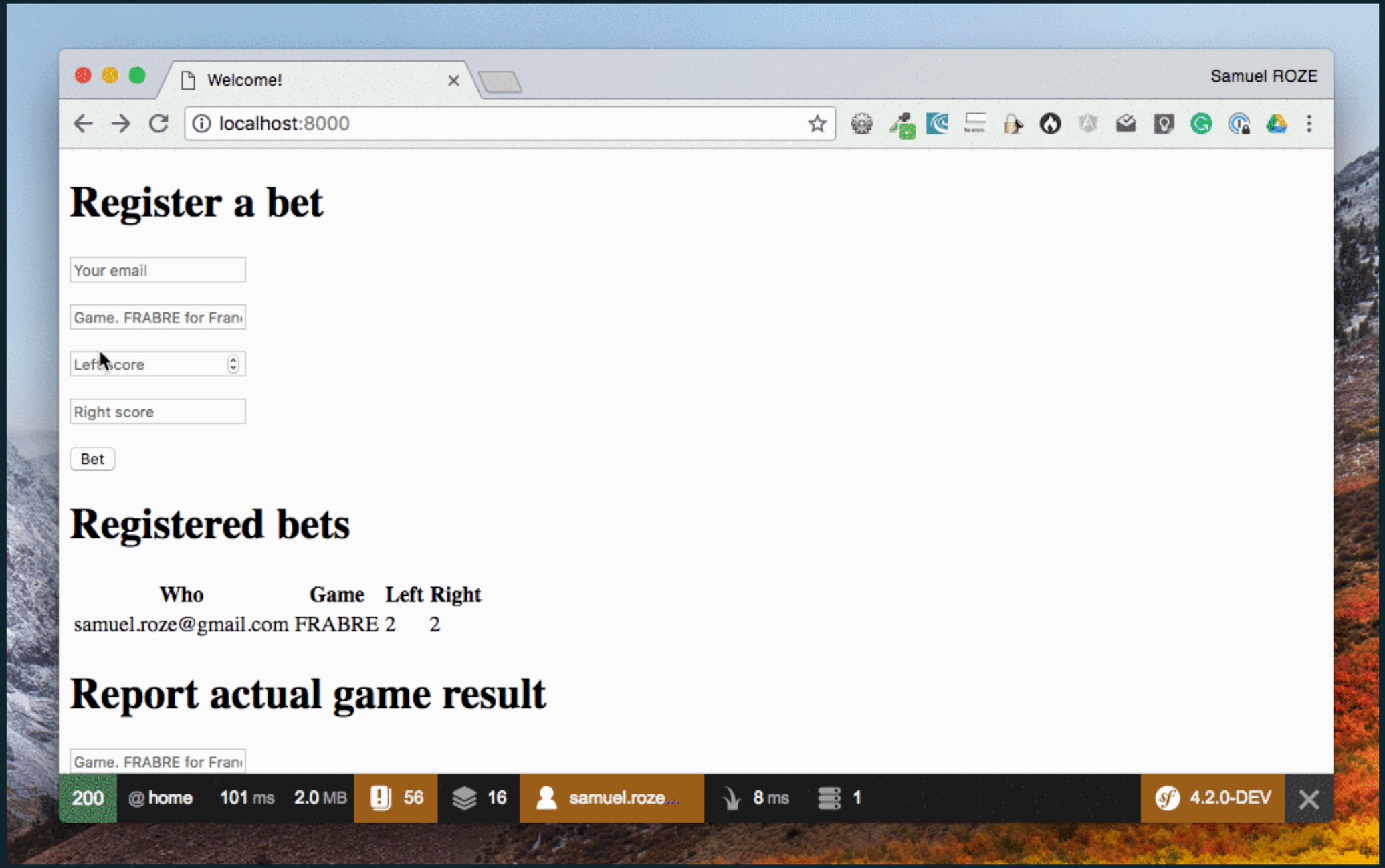

RegisterBet
dispatch
ReportGameResult
CommandBus
handles
Handlers
GetAllBets
dispatch
WonBet
Handlers
EventBus
LostBet


Welcome!
Samuel ROZE
localhost:8000
Register a bet
Your email
Game. FRABRE for Fran
Left score
Right score
Bet
200
@ home
1024 ms
8.0 MB
56
135
anon.
21 ms
4.2.0-DEV

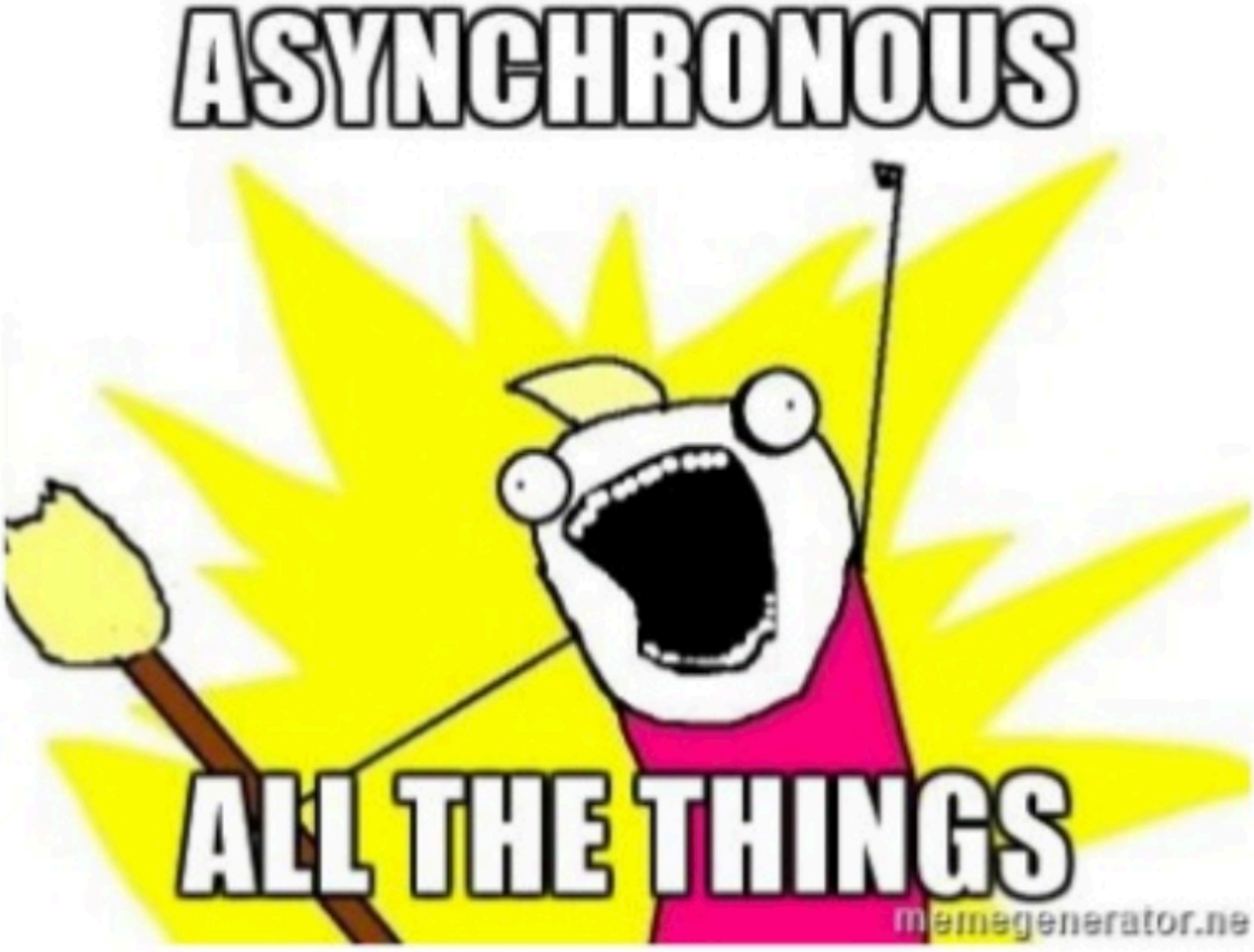

Welcome!
Samuel ROZE
localhost:8000
Register a bet
samuel.roze@gmail.com
FRABRE
2
2
Bet
200
@ home
38 ms
2.0 MB
53
4
anon.
5 ms
4.2.0-DEV


Welcome!
Samuel ROZE
localhost:8000
Register a bet
Your email
Game. FRABRE for Fran
Left score
Right score
Bet
Registered bets
Who Game Left Right
samuel.roze@gmail.com FRABRE 2 2
200 @ home 2253 ms 10.0 MB 55 78 samuel.roze... 13 ms 1 4.2.0-DEV


Welcome!
Samuel ROZE
localhost:8000
Game. FRABRE for Fran
Left score
Right score
Bet
Registered bets
Who Game Left Right
samuel.roze@gmail.com FRABRE 2 2
Report actual game result
Game. FRABRE for Fran
Left score
Right score
Bet
200 @ home 2147 ms 12.0 MB 55 83 samuel.roze... 11 ms 1 4.2.0-DEV


Welcome!
Samuel ROZE
localhost:8000
Register a bet
Your email
Game. FRABRE for Fran
Left score
Right score
Bet
Registered bets
Who Game Left Right
samuel.roze@gmail.com FRABRE 2 2
Report actual game result
Game. FRABRE for Fran
200 @ home 101 ms 2.0 MB 56 16 samuel.roze... 8 ms 1 4.2.0-DEV

```
docker run -d --hostname rabbit --name rabbit -p
15672:15672 -p 5672:5672 rabbitmq:3-management
```


ASYNCHRONOUS
ALL THE THINGS
memegenerator.ne

https://github.com/sroze/messenger-workshop

More?